

Taller primeros pasos con Piral (Hola Mundo)

Creación de un mf - tipo Shell(piral):

1. Crear un directorio y en esa ruta ejecutar:
“piral new --target piral-shell”
De ser necesario instalar los paquetes requeridos

```
C:\Users\alansdor\Documents\CelulaFront\POC_Piral\Ejemplo>piral new --target piral-shell
+ piral-cli v1.6.2
✓Piral instance scaffolded successfully!
```

Nota: Puede cambiar “piral-shell” por el nombre que le quiera dar al Shell.

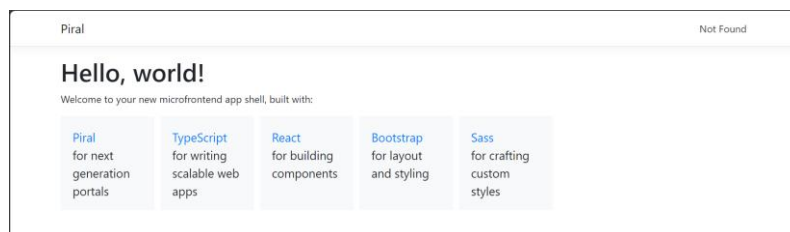
2. Ejecutar en ambiente de desarrollo:
 - “cd piral-shell”
 - “npx piral debug” o “npx piral debug --port 4200” para especificar el puerto de subida
 - Debe seleccionar un bundler, escoger el de su preferencia

```
C:\Users\alansdor\Documents\CelulaFront\POC_Piral\Ejemplo>cd piral-shell

C:\Users\alansdor\Documents\CelulaFront\POC_Piral\Ejemplo\piral-shell>npx piral debug
+ piral-cli v1.6.2
? Reading configuration ...? No bundler found. Which one do you want to install? (Use arrow keys)
> webpack5
  vite
  vite5
  xbuild
  bun
  esbuild
  parcel
! Reading configuration ...

C:\Users\alansdor\Documents\CelulaFront\POC_Piral\Ejemplo\piral-shell>npx piral debug
+ piral-cli v1.6.2
? No bundler found. Which one do you want to install? vite
!> Running at http://localhost:1234/
!> Manage via http://localhost:1234/manage-mock-server
vite v4.5.5 building for production...
```

- Acceder a <http://localhost:1234> (ruta por defecto) y el resultado será:



Nota: Ya tenemos un “Hello World”, pero no hemos integrado Microfrontends.

- “npm run build”, esto generará la carpeta dist/emulator

```
└─ piral-shell
  └─ dist
    └─ emulator
      └─ piral-shell-1.0.0.tgz
```

Creación de un mf - tipo App(pilet)

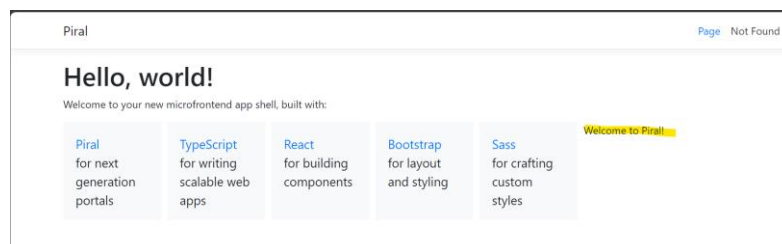
Importante: Es obligatorio tener un Shell(piral)

1. En la carpeta raíz del proyecto ejecutar:
“`piral new ./piral-shell/dist/emulator/piral-shell-1.0.0.tgz --target pilet-header`”
Nota: Debe cambiar “piral-shell” por el nombre que le dio al Shell al momento de crearlo y puede cambiar “pilet-header” por el nombre que le quiera dar al Microfrontend.
2. Ejecutar en ambiente de desarrollo:
 - “`cd pilet-header`”
 - “`npx pilet debug`” o “`npx pilet debug --port 4210`” para especificar el puerto de subida
 - Debe seleccionar un bundler, escoger el de su preferencia

```
C:\Users\alansdor\Documents\CelulaFront\POC_Piral\Ejemplo\pilet-header>npx pilet debug
+ piral-cli v1.6.2
? Reading configuration ...? No bundler found. Which one do you want to install? (Use arrow keys)
> webpack5
  vite
  vite5
  xbuild
  bun
  esbuild
  parcel
? Reading configuration ...

C:\Users\alansdor\Documents\CelulaFront\POC_Piral\Ejemplo\piral-shell>npx piral debug
+ piral-cli v1.6.2
? No bundler found. Which one do you want to install? vite
!> Running at http://localhost:1234/
!> Manage via http://localhost:1234/manage-mock-server
vite v4.5.5 building for production...
```

- Acceder a <http://localhost:1234> (ruta por defecto) y el resultado será:



Note que al desplegar este Microfrontend tenemos como base el contenido completo del Shell más un nuevo texto que dice “Welcome to piral”, este es el contenido de nuestro Microfrontend.

Adicional: En el archivo `index.tsx` del Shell encontrará la url del feed, de la cual puede cambiar la palabra “empty” por “sample” y podrá notar que cambia el consumo de Microfrontends.

```
index.tsx X
piral-shell > src > index.tsx > ...
1 import * as React from 'react';
2 import { createRoot } from 'react-dom/client';
3 import { createInstance, Piral, createStandardApi } from 'piral';
4 import { layout, errors } from './layout';
5
6 // change to your feed URL here (either using feed.piral.cloud or your own)
7 const feedUrl = 'https://feed.piral.cloud/api/v1/pilet/empty';
```

Cada línea de código es un paso más hacia el futuro que imaginas. ¡Sigue creando!